Rapid co-optimisation of turn-on and turn-off gate resistor values in DC:DC power converters

Harry C. P. Dymond, Jeremy J. O. Dalton, and Bernard H. Stark
Electrical Energy Management Group, Faculty of Engineering, University of Bristol, Bristol, UK

***Abstract* — Both turn-on and turn-off gate resistance for switched power devices can significantly impact system EMI, switching loss, and device longevity. To determine the optimum resistor values, an exhaustive search through *m* different turn-on values and *n* different turn-off values would require $m \times n$ tests. This paper demonstrates a method that separates this process into $m+n-1$ experimental tests, followed by analysis of measured losses and time-domain waveforms in MATLAB. The methods allow loss and waveform spectral content to be highly accurately predicted for any of the possible $m \times n$ resistor combinations. Losses are predicted using 2D curve fitting, whilst automated edge-extraction and splicing is used on the time-domain waveforms for subsequent spectral analysis. Significant time savings are delivered through the reduction of repeated reworking to swap gate resistances, reconnection to test equipment, and the waiting for the converter to reach thermal steady state. All of the MATLAB code is made freely available to readers.**

***Keywords* — Edge Detection, Edge Extraction, Waveform Splicing, Waveform Reconstruction, Gate Resistor Optimisation**

I. Introduction

A key design aspect for tuning the performance of power electronic converters is the value of resistance used at the gate of power devices. Many gate drivers offer independent turn-on and turn-off paths (Fig. 1*a*) [1] – [3] and for those that don't, simple networks external to the gate driver (Fig. 1*b*) can offer asymmetrical turn-on and turn-off gate-drive resistance.

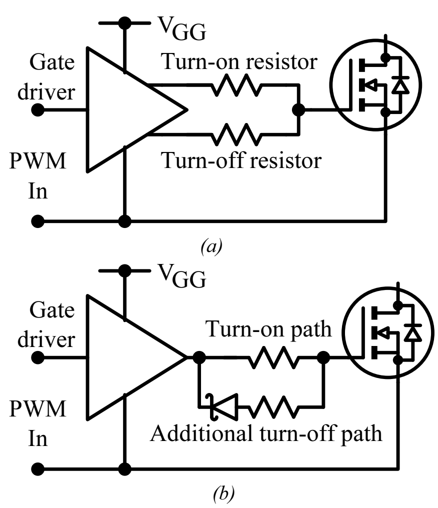

Fig. 1 Gate driving with independent turn-on and turn-off resistance. *(a)* Using a driver with independent pull-up and pull-down outputs, *(b)* Using networks external to the gate driver.

This work was supported by the U.K. Engineering and Physical Sciences Research Council (EPSRC) under grants EP/K021273/1 and EP/K034804/1

Independent turn-on and turn-off gate drive resistance gives designers maximum flexibility, allowing independent adjustment of the controlled device's turn-on and turn-off behaviour. When selecting the values of gate drive resistance, the designer will typically have to trade-off power losses against other performance factors such as spectral content of voltage and current switching waveforms (for electromagnetic compatibility), overshoots, and ringing. The optimum values of gate resistance are usually determined experimentally [4],[5] rather than via simulation, due to a number of simulation challenges:

1. Simulation models can require a large investment of person-hours to deliver an acceptable match to reality, requiring experimental validation.
2. The execution time of accurate circuit models can be very high.

Taken together, the total time taken to perform an evaluation of gate drive resistances in simulation can exceed that of performing it experimentally. An exhaustive evaluation of gate drive resistance combinations would require $m \times n$ experiments for *m* different turn-on resistances and *n* different turn-off resistances. For each change in resistor value, the circuit under test must be disconnected from test equipment, have a resistor removed and replaced with another, be re-connected to the test equipment, and be operated until it reaches a thermal steady-state. Not only is this a lengthy process, the repeated reworking of the circuit board to remove and replace the resistors can eventually lead to irrecoverable physical damage such as lifted solder pads and obliterated solder resist.

Even for small *m,n*, the time required to perform a complete exhaustive evaluation can be prohibitive, forcing measurements over a smaller solution space and thereby potentially missing the optimum value combination. This paper presents methods to allow an accurate, exhaustive performance evaluation across all $m \times n$ resistor value combinations, whilst only requiring $m+n-1$ experiments. This significant reduction in experimentation time for a given solution space raises the possibility of expanding the number of candidate turn-on and turn-off resistors, to give confidence that the best combination of resistor values will be found. Optimum resistor values may be determined for a given target system performance, such as minimisation of losses for a given maximum acceptable spectral content in voltages and/or currents in the converter.

For each experiment, power-circuit time-domain waveforms are acquired and saved, together with measured power losses. The power losses for any of the $m \times n$ resistor value combinations can be predicted via standard 2D curve fitting [6]. An overview of the automated process of the time-domain data analysis by an edge extraction and splicing method is provided in Section II. At present, the method relies on the switch-mode waveforms being steady-state, such as those found in a DC:DC converter when input and output conditions are not changing, but similar methods could be used in an enhanced scheme with time-varying switching waveforms such as those found in an inverter. This paper presents the methods used for:

- The automated switching-edge detection and extraction, which minimises the amount of leading data extracted before the start of the edge, and maximises the amount of trailing data extracted following the edge, as explained in Section III.
- The automated edge splicing method used to build a "predicted" time-domain waveform for the desired turn-on and turn-off resistor combination, as detailed in Section IV. The method avoids introducing discontinuities in the reconstructed waveform that would otherwise corrupt its high-frequency content.
- The reconstruction of multiple different time-domain waveforms from a single circuit (i.e. voltages and currents at various points in the circuit) including non-switching waveforms that exhibit disturbances (for example the output of a converter can exhibit disturbances when the power devices switch), as detailed in Section V. The reconstruction maintains the correct relative timing of all waveforms.

Once time-domain waveforms have been constructed, they can be transformed into the frequency domain for spectral analysis, and these data subsequently used for the gate resistor value co-optimisation. In Section V, the results of using the edge extraction and splicing methods on waveforms acquired from a SiC-based boost converter are presented, demonstrating that the spectral content of reconstructed waveforms, and predicted system loss, closely match actual measurements.

In Section VII, the reconstructed waveforms and interpolated losses are used to automatically determine the optimum gate resistor value combination, based on a set of user-defined selection criteria such as an acceptable limit for the spectral envelopes of switching waveforms.

Section VII draws conclusions, noting some limitations of the present implementation and making suggestions for further work.

II. Overview of Experimental Procedure and Subsequent Edge Extraction and Splicing

For this paper, the method is used on an open-loop 300 W, 600 V output non-synchronous 1:10 SiC boost converter operating with 100 kHz switching frequency, as shown in Fig. 2. Time-domain waveforms of v_{GS}, v_{SW} and v_{OUT} are captured on a 10 GSa/s, 4 GHz Rhode & Schwarz RTO1044 oscilloscope, and input and output power (for loss calculation) are measured using three 6.5-digit Fluke 8845A multimeters (for V_{IN}, V_{OUT}, and I_{OUT}) and one 6.5 digit GW PCS-1000 precision current shunt (for I_{IN}). It is desired to optimise the gate resistor values over a range of 2.7 Ω to 47 Ω for turn-on, and 2.7 Ω to 100 Ω for turn-off, to find the best compromise between losses and spectral content of the measured v_{SW} and v_{OUT} waveforms. The first step of the process is to acquire measurement data for the turn-on and turn-off resistor value combinations shown in Table I. 11 turn-off and 9 turn-on resistor values are evaluated with 19 tests. Whilst all resistor values could be used in a minimum of 11 tests (by simultaneously changing both turn-on and turn-off resistor values), this would give too-small a measurement space to go into the 2D loss curve fitting, and adversely affect the accuracy of loss predictions for untested resistor combinations.

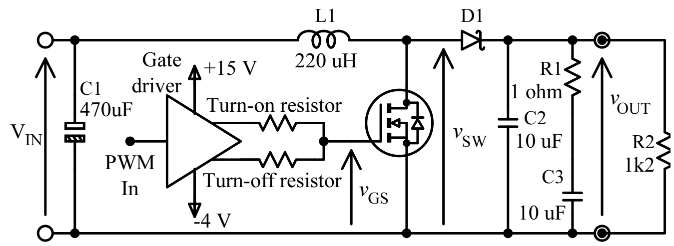

Fig. 2 Schematic of boost converter used in this work to demonstrate the analysis methods.

TABLE I. Turn-on and turn-off resistor value combinations used when acquiring circuit measurements

Turn-on resistor value (Ω)	Turn-off resistor value (Ω)
10	100
10	68
47	47
47	33
47	22
47	15
47	10
47	6.8
47	4.7
47	3.3
47	2.7
33	10
22	10
15	10
10	10
6.8	10
4.7	10
3.3	10
2.7	10

For each resistor combination, the converter is operated until it reaches thermal steady state, at which point losses and time-domain signals are measured and acquired. The oscilloscope is set to capture and average multiple successive waveforms to give a high signal to noise ratio, with each capture duration equal to exactly one complete switching cycle (i.e. 1/100 kHz = 10 µs). Triggering is set so that the switching edges occur well away from the start and end points of the capture. Loss measurements and time-domain waveforms are transferred to a host PC for offline analysis in MATLAB.

To determine the time-domain waveforms that would occur for any gate drive resistor combination that is not part of the measurement space, e.g. 6.8 Ω turn-on with 33 Ω turn-off, the procedure outlined in Fig. 3 is followed. Firstly, edges are extracted from the relevant measured data (in this case the waveforms for 47 Ω on with 33 Ω off, and 6.8 Ω on with 10 Ω off) and then spliced to form a new "reconstructed" waveform.

Because the source data from each edge has been captured under the influence of minor test-to-test variations, and due to

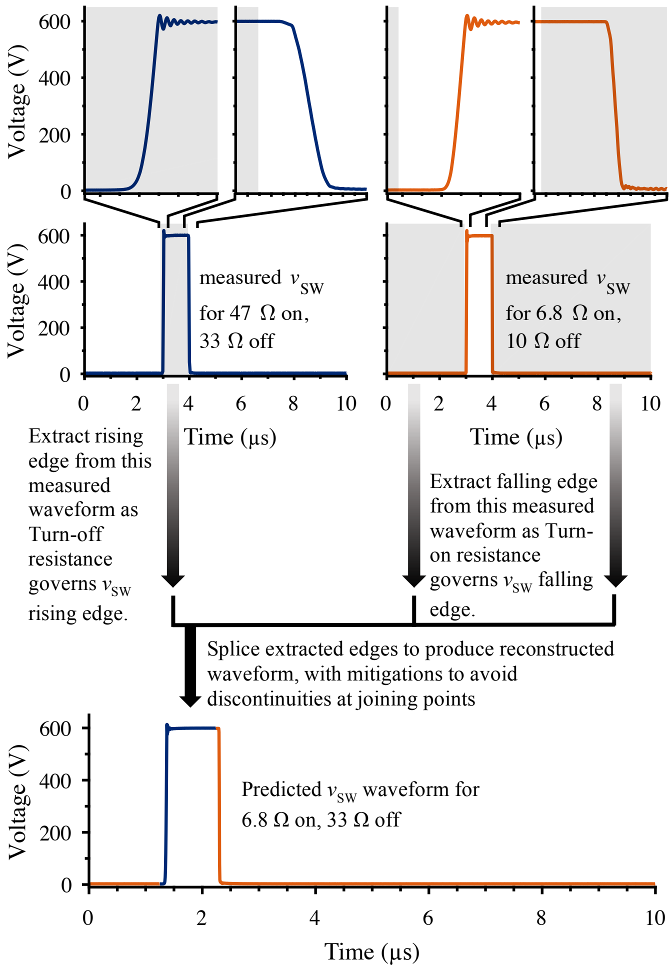

Fig. 3 Overview of edge extraction and splicing process, for predicting v_{SW} waveform with 6.8 Ω turn-on resistance and 33 Ω turn-off resistance. The top graphs are horizontal zooms of the two "source" waveforms, showing the rising and falling edges in more detail.

small time- and temperature- dependent variations in the inaccuracies of the waveform acquisition, the high and low steady-state values of the edges to be spliced will not be exactly equal. The splicing process must account for this, so as not to introduce discontinuities which would corrupt the high-frequency content of the reconstructed waveforms.

III. Edge Extraction Method

A. Sliding-window Linear Regression

To extract edges from a measured waveform, first the vertical midpoint of the waveform is found:

```
midpoint = (max(timeData)-min(timeData))/2;
```

The first edge is located by searching forwards from the start of the waveform, looking for the first index where the data crosses the `midPoint` value. For example, if the initial waveform value is below `midPoint`, the index of the transient of the first edge is located at:

```
firstEdgeIndex = find(timeData>midPoint,1);
```

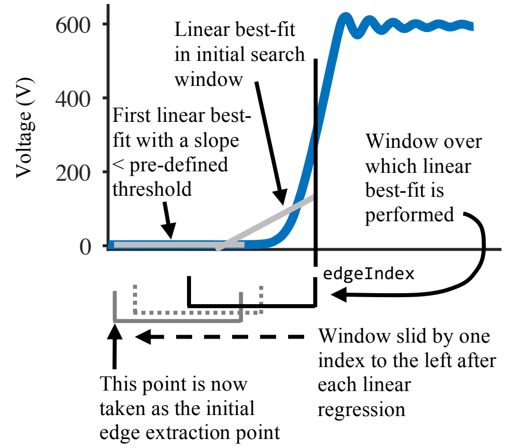

Fig. 4 Linear regression performed over a sliding window to find waveform steady-state.

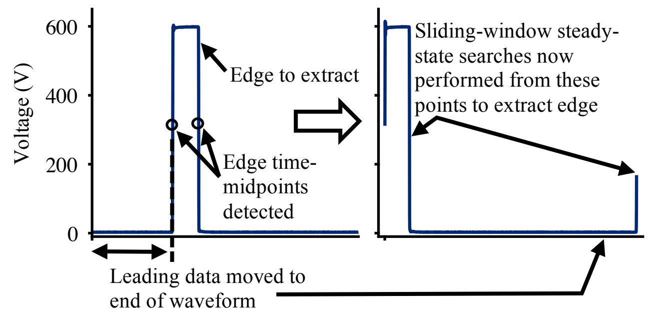

Fig. 5 Waveform "shuffling" prior to edge extraction if edge to be extracted is the second edge.

and the index of the second edge's transient is located by searching backwards from the end of the waveform:

```
secondEdgeIndex = find(timeData>midPoint,1,'last');
```

If the edge to be extracted is the first edge, the start of its transient must be located; this will define the initial edge extraction point. The final edge extraction point is found by searching backwards from the second edge transient, looking for the point where that transient starts.

The starts of transients are found using the process illustrated in Fig. 4. A sliding window, equal in length to 0.5% of the total switching period, defines the region over which a linear regression is performed on the data. Initially, the window is placed at the relevant `edgeIndex` point, and is then slid backwards one index at a time, once the linear regression has been performed. The process is repeated until the slope of the best-fit line in the window is below a fixed threshold, indicating that a steady-state has been located.

If the edge to be extracted from the data is the second edge, the waveform is first "shuffled", as shown in Fig. 5, before performing the steady-state searching process to find the start and end points for the edge extraction.

B. Alternative Edge Extraction Method

Edges could potentially be identified using a high-pass filtering and smoothing process depicted in Fig. 6. Here, the

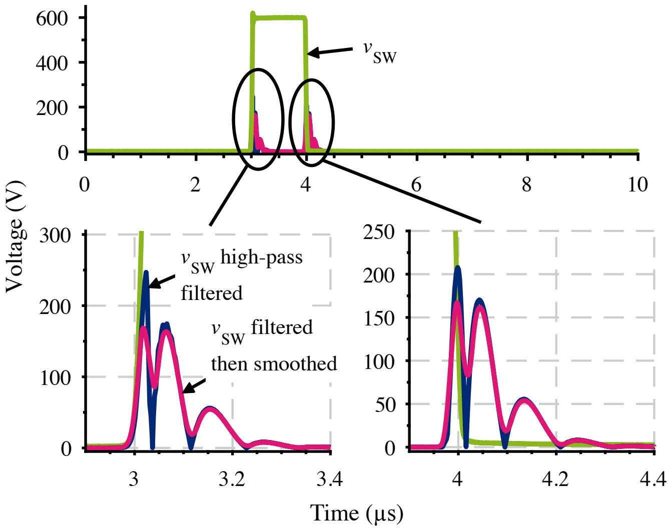

Fig. 6 Edge detection with high-pass filtering and smoothing.

switching waveform has been processed by a high-pass filter with a stopband (80 dB attenuation) width of 250 kHz and a passband (0 dB ± 1 dB) starting at 10 MHz. The absolute values of the filtered waveform are then passed through a 255-point moving-average filter to give the smoothed line in Fig. 6. If the value of this line is above a pre-determined threshold, the source waveform can be considered to be undergoing a transition.

Although this edge-detection method can be faster than the sliding-linear-regression method, both methods are fast – taking the order of tens of milliseconds to detect edge start and end points in a 100,000-point waveform. The linear regression method also requires only two parameters to be configured: the sliding window length and a "flatness" threshold. The filtering method requires the high-pass and moving-average filter characteristics to be tuned to the input waveforms, definition of a threshold for the smoothed filtered waveform to define when a transition is occurring, and a definition of the number of samples that should be extracted prior to the detected edge (unlike the linear regression method, the filter method will not find a long enough steady-state prior to the edge such that the extracted edges are ready for splicing). The linear regression method was therefore selected as the edge detection method in this work.

IV. Edge Splicing Method

When the edges are to be spliced together to form a "reconstructed" time-domain waveform, it is vital that discontinuities are not introduced, as these would create spurious high-frequency content. However, because the source data for the edges to be spliced are acquired under the influence of minor test-to-test variations, and due to small time- and temperature- dependent variations in the inaccuracies of the equipment, if the waveform high-level amplitudes (e.g. the 600 V level of the example v_{SW} waveforms shown in Fig. 3) are aligned, there may still be a discontinuity at the low level. The edge-alignment conflict is handled by aligning the pulses at the steady-state value found in-between the two edges (e.g. 600 V for the example v_{SW} waveforms), and then windowing the data to accommodate any discontinuity in the waveform end-points. The pulse-alignment process follows the procedure illustrated in Fig. 7, and the windowing process follows the procedure illustrated in Fig. 8.

When the process is complete, the waveform gently tapers to/from equal values at the start and end of the waveform, ensuring no discontinuity is present when the frequency content is calculated via the FFT.

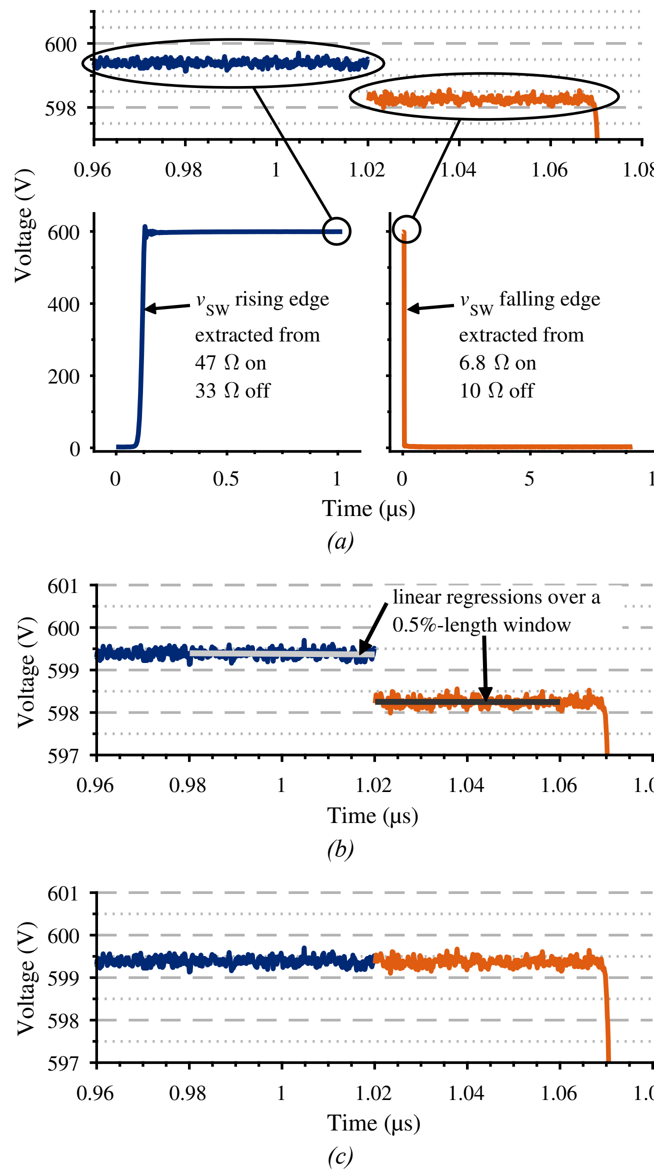

Fig. 7 Pulse alignment process

(a) Discontinuity between end of rising edge and start of falling edge must be accommodated.

(b) Linear regressions over a 0.5%-length window calculated at end of first-edge data and start of second-edge data.

(c) DC offset of the second edge data adjusted to align the initial value of the second-edge best-fit line, to the final value of the first-edge best-fit line.

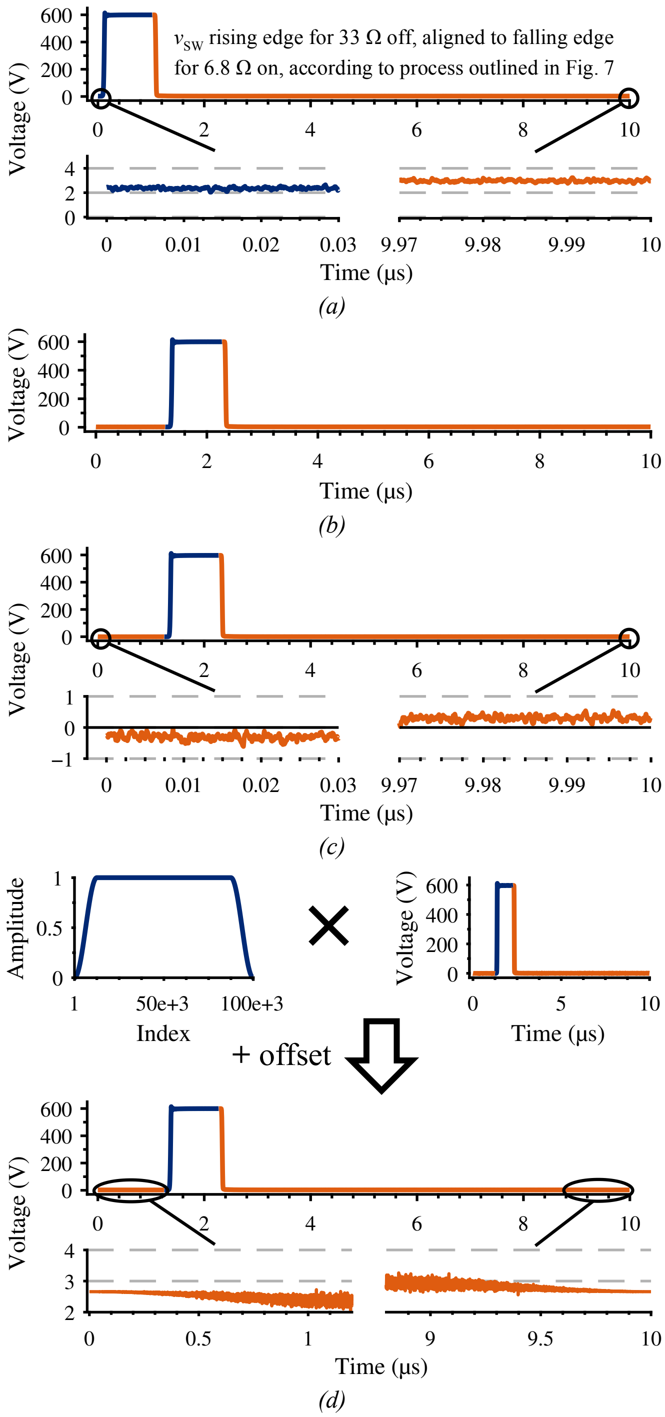

Fig. 8 Edge splicing windowing process

(a) Discontinuity between end of falling edge and start of rising edge must be accommodated.

(b) Sufficient quantity of data moved from the end of the waveform to the start, to guarantee that neither edge will be affected by the window to be applied. The offset of the moved data is adjusted to align them to the start data.

(c) Average of the DC offsets at the start and end of the waveform subtracted from the waveform.

(d) 0.25-taper Tukey (tapered cosine) window applied to data, and DC offset subtracted in step *(c)* added back.

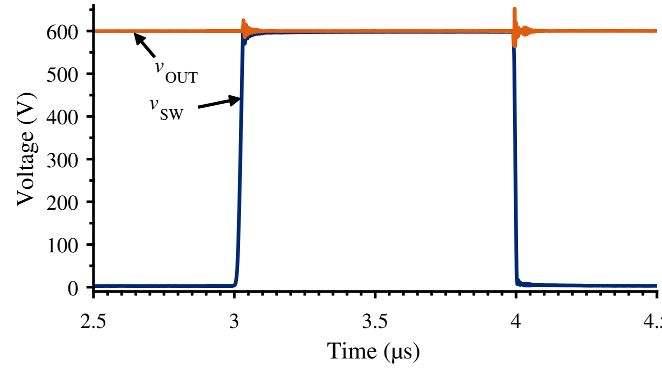

Fig. 9 v_{OUT} disturbances during power device switching for the example circuit of Fig. 2.

V. Reconstruction of Multiple Waveforms

A. Overview

Often there will be multiple waveforms of interest in a power converter. The techniques presented can be applied to multiple waveforms, but additional processing is required to ensure that predicted waveforms maintain the correct relative timing. Additionally, some waveforms may not be "switching" waveforms like the v_{SW} waveform used thus far as an example, but may instead exhibit disturbances when a power device switches. For the circuit used in this work, an example of one such "disturbance" waveform is the output voltage (marked v_{OUT} in Fig. 2) of the converter. Whenever the power FET switches, the output voltage momentarily deviates from its DC value, as shown in Fig. 9.

B. Reconstruction of Disturbance Waveforms

As a disturbance waveform is non-switching, the sliding-window linear regression method described in Section III.A cannot be used on such waveforms. Instead, a switching waveform is used as a "reference", whereby the start and end indices for edges extracted from the source reference waveforms are saved during the edge extraction process, and are then used to define the indices at which data should be extracted from the source disturbance waveforms. Once fragments have been extracted from the source disturbance waveforms, they are spliced together using the process described in Section IV. It must be noted that this method requires that the disturbances do not occur before the transients in the reference switching waveform. In the example provided here, v_{SW} is a suitable candidate to act as the reference, because the v_{OUT} disturbances are occurring as a direct result of the v_{SW} transients and therefore cannot occur before them.

If no switching waveforms are available to act as a reference, disturbances could be detected using a method based on that described in Section III.B. However, it is most probable that the switching waveform responsible for the disturbances would be of interest and therefore would be measured and available to act as a reference. In this case, extracting the disturbances from non-switching waveforms is very rapid as no additional computation is required to determine the extraction points.

C. Maintaining Synchronisation of Multiple Switching Waveforms

When reconstructing multiple switching waveforms, one of the switching waveforms acts as a reference. Edges are extracted from the reference waveform first, with the index of the vertical midpoint of each edge being saved for later reference. Edges are then extracted from the other switching waveforms, and during this process the time delay to the reference waveform edges are calculated and saved, as illustrated in Fig. 10. Reconstruction of the waveforms then proceeds as follows:

1. The reference waveform is reconstructed according to the steps outlined in Section IV, with the additional proviso that the amount of data moved from the end of the waveform to the start (as per Fig. 8*(b)*) is now based on the earliest-occurring edge across all the switching waveforms. This is because all reconstructed waveforms must have sufficient lead-in data such that no edge will be affected by the Tukey windows that will be applied later to accommodate any end-to-start discontinuities. For the example v_{SW} and v_{GS} waveforms, the v_{GS} falling edge starts before the v_{SW} rising edge, as shown in Fig. 10. So, in this case, the amount of lead-in data required is determined by the time position of the v_{GS} falling edge.

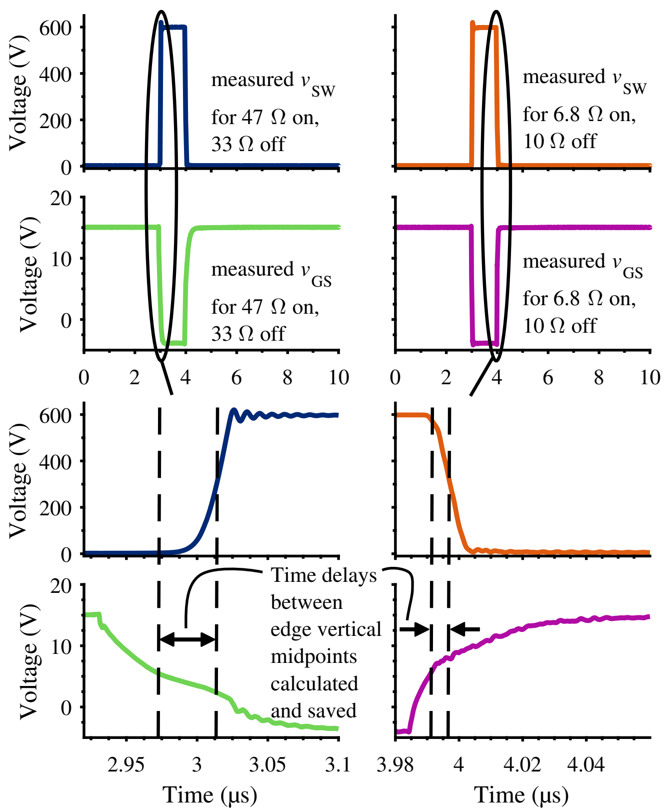

Fig. 10 Maintaining synchronisation across multiple predicted switching waveforms: during edge extraction process, the time delays between the measured switching edges in the soure waveforms are calculated and saved. v_{SW} and v_{GS} switching edges for circuit of Fig. 2 used as an example here.

2. The edges of any non-reference switching waveforms are now positioned according to the time-delays saved during the edge extractions, as shown in Fig. 11.
3. For each of the non-reference switching waveforms, the end of the first edge may now overlap with the start of the second edge, or there may be a gap between the edges such as shown in Fig. 11, bottom. If there is overlap, data are removed from the first edge. If there is a gap, it is filled by repeating data from the end of the first edge.
4. The pulse amplitudes of each of the non-reference switching waveforms are aligned according to the steps shown in Fig. 7.
5. For each of the non-reference switching waveforms, those that are too long are truncated, and those that are too short are extended by repeating data at the end of the second edge.
6. Data are moved from the end of the waveforms to the start, as illustrated in Fig. 12.
7. The waveforms are windowed according to the steps shown in Fig. 8 *(c)* and *(d)*.

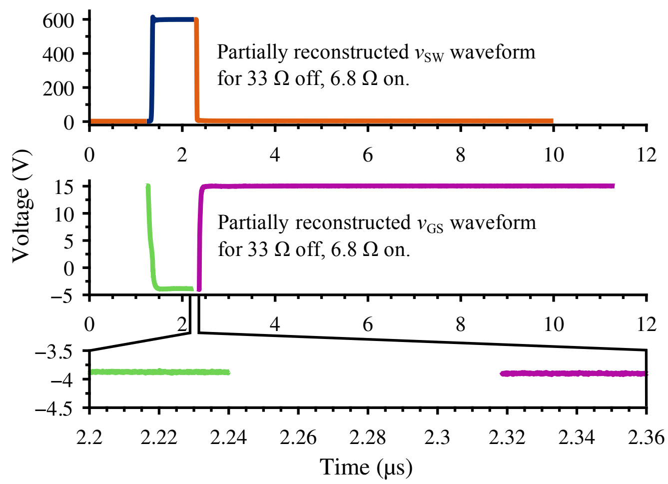

Fig. 11 Edges of non-reference waveforms, v_{GS} in this case, positioned to maintain the previously saved time delays. The bottom graph shows a horizontal zoom of the middle graph.

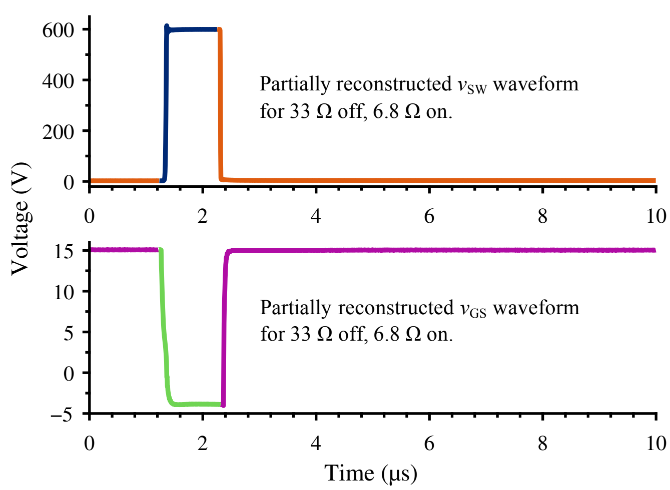

Fig. 12 Pulses aligned, and data moved from end of waveform to start, ready for the final stage of reconstruction: windowing as per Fig. 8.

VI. Comparison of the Frequency-Domain Content of Predicted and Measured Waveforms For a SiC-based Boost Converter

The presented waveform analysis methods have been used to reconstruct v_{SW} and v_{OUT} waveforms for two resistor combinations not covered by those listed in Table I: 6.8 Ω on with 33 Ω off, and 33 Ω on with 33 Ω off. Also, 2D curve fitting is used to predict the losses for these resistor combinations. Subsequently, the real circuit has been operated with these two additional resistor value combinations, to validate the analysis method. The loss comparison is provided in Table II. In both cases, the error between predicted loss and measured loss is within the measurement error band.

TABLE II. Comparison between predicted and measured losses for two turn-on/turn-off resistor combinations

Turn-on resistor value (Ω)	Turn-off resistor value (Ω)	Predicted loss (W)	Measured loss (W)	Error (W, (%))
6.8	33	14.50	14.23	0.27 (1.9%)
33	33	16.45	16.32	0.13 (0.8%)

Fig. 13 through Fig. 16 show comparisons between the spectral envelopes for v_{SW} and v_{OUT}, calculated from the time-domain waveforms. In all cases, the spectral content of the predicted waveforms matches closely to the actual measured waveforms.

VII. Optimising the Gate Resistors for Best Loss vs. Spectral Content Tradeoff

Using the presented methods, MATLAB can be used to search across all $m \times n$ gate resistor value combinations, calculating predicted losses and circuit waveform spectral content. If acceptable spectral content limits are defined, it can then be determined which gate resistor value combination meets the defined limit with the minimum losses.

For example, if the v_{OUT} and v_{SW} spectral content limits are defined as in Fig. 17 and Fig. 18 respectively, it is determined

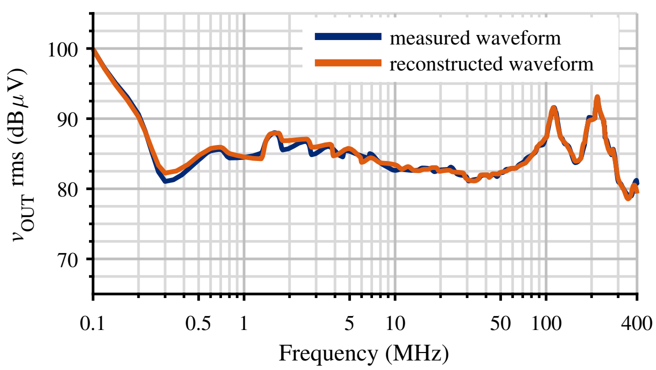

Fig. 13 Predicted and measured v_{OUT} spectral envelopes for 6.8 Ω turn-on and 33 Ω turn-off.

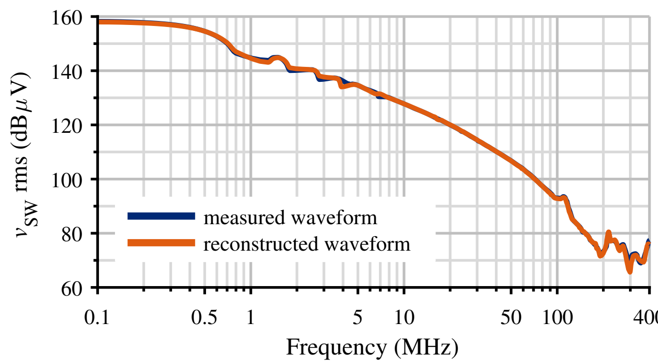

Fig. 14 Predicted and measured v_{SW} spectral envelopes for 6.8 Ω turn-on and 33 Ω turn-off.

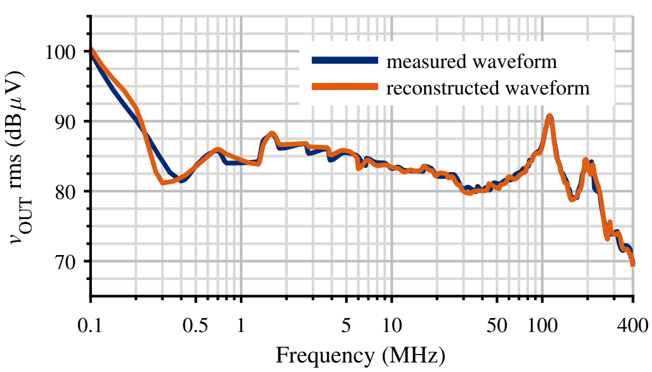

Fig. 15 Predicted and measured v_{OUT} spectral envelopes for 33 Ω turn-on and 33 Ω turn-off.

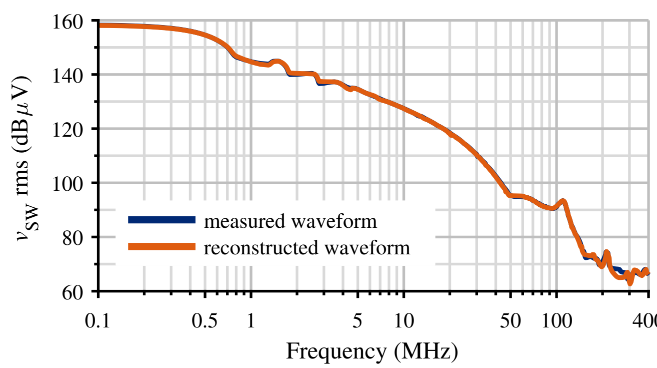

Fig. 16 Predicted and measured v_{SW} spectral envelopes for 33 Ω turn-on and 33 Ω turn-off.

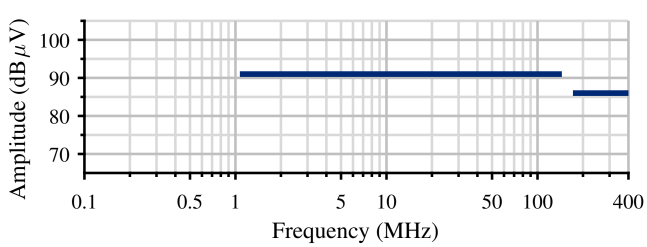

Fig. 17 An example spectral limit definition for v_{OUT}.

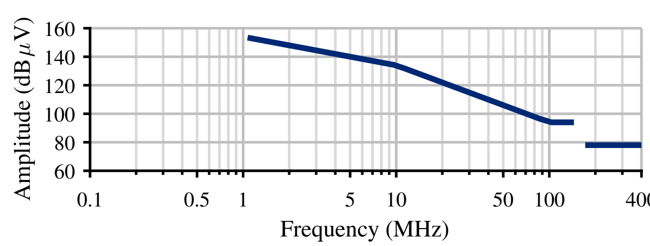

Fig. 18 An example spectral limit definition for v_{SW}.

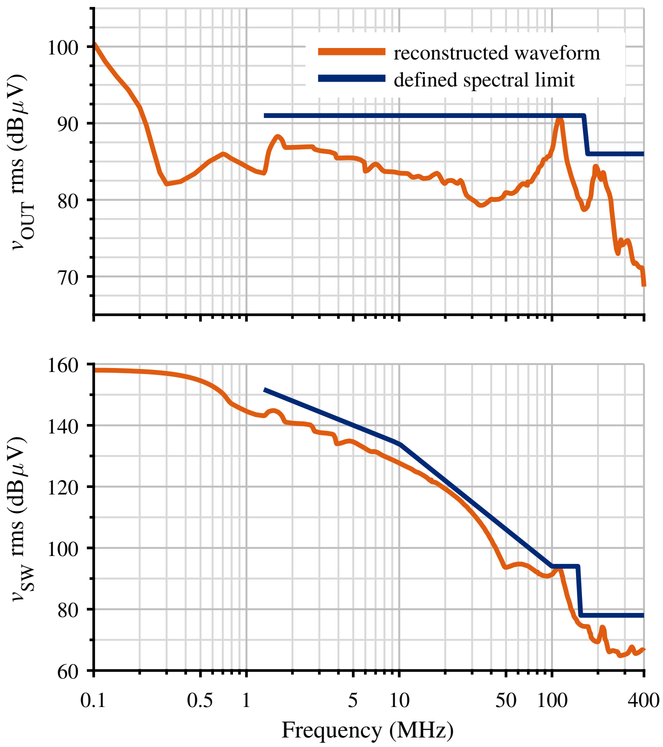

Fig. 19 Predicted v_{OUT} and v_{SW} spectral envelopes for 33 Ω on, 6.8 Ω off.

that 19 different combinations of turn-on and turn-off resistor values meet these limits. One of these combinations is 33 Ω turn-on with 6.8 Ω turn-off, as shown in Fig. 19. With power circuit losses of 15.51 W, this is the lowest-loss resistor combination to meet the limit. The total execution time of the optimisation is 3.3 seconds when run on a 2.6 GHz Intel Core i3 system with 16 GiB RAM running MATLAB R2016a.

VIII. Conclusions and Further Work

Automated methods for extracting and splicing edges from measured time-domain waveforms have been presented. These form the key part of a process for determining the optimum values of a power device's turn-on and turn-off gate drive resistances with a minimum number of experiments required. Exhaustive evaluation of all of the $m \times n$ resistor value combinations, for m different candidate turn-on values and n candidate turn-off values, is possible whilst only requiring $m+n-1$ experiments. Once suitably-formatted data have been experimentally acquired, the analysis is highly automated and extremely rapid.

Currently, the analysis methods are only applicable to DC:DC converters with fixed operating conditions, but the concept could potentially be expanded to accommodate variable conditions such as changing load current. Switching waveforms must have a pulse duration of at least 0.5% of the switching cycle, and noisy waveforms or those that do not reach a steady-state between edges, for example a waveform with extensive post-edge ringing, are not currently supported.

The edge extraction and splicing methods presented have other potential uses beyond gate resistor optimisation. They could form the basis of experimentally-informed circuit simulation, whereby a database of measured waveforms for a range of operating conditions is created using a test circuit, and then the simulation tool extracts and splices edges from the database to create simulated waveforms for any scenario.

MATLAB code that implements the presented methods is available at [7], together with a set of example data.

References

[1] Maxim Integrated, "MAX5048C data sheet," [online: https://datasheets.maximintegrated.com/en/ds/MAX5048C.pdf]

[2] Silicon Labs, "Si8271 data sheet," [online: https://www.silabs.com/documents/public/data-sheets/Si827x.pdf]

[3] Texas Instruments, "LM5114 data sheet," [online: http://www.ti.com/lit/ds/symlink/lm5114.pdf]

[4] B. Sun, R. Burgos, X. Zhang and D. Boroyevich, "Active dv/dt control of 600V GaN transistors," *2016 IEEE Energy Conversion Congress and Exposition (ECCE), Milwaukee*, WI, 2016, pp. 1-8.

[5] J. Gottschlich and R. W. De Doncker, "A programmable gate driver for power semiconductor switching loss characterization," *2015 IEEE 11th International Conference on Power Electronics and Drive Systems*, Sydney, NSW, 2015, pp. 456-461.

[6] The MathWorks Inc., "Fit curve or surface to data," [online: http://www.mathworks.com/help/curvefit/fit.html]

[7] University of Bristol, MATLAB waveform splicing tools, [online: http://tiny.cc/waveformSplicing]